

A

06/15/99

JCS18 U.S. PTO  
09/333855  
06/15/99

<b>UTILITY PATENT APPLICATION TRANSMITTAL</b>  <small>(Only for new nonprovisional applications under 37 CFR 1.53(b))</small>	<i>Title of Invention</i>	Safe Save Method for HTML Documents
	<i>Named Inventor(s)</i>	Alan A. Ramaley, Thomas W. Olsen, Darrell L. Aldrich and David M. Buchthal
	<i>Attorney Docket</i>	13237-2390
	<i>Express Mail Label No.</i>	EL329859111US

<b>APPLICATION ELEMENTS</b>	Assistant Commissioner for Patents <b>ADDRESS TO:</b> Box Patent Application Washington, D.C. 20231
<ol style="list-style-type: none"> <li><input type="checkbox"/> Fee Transmittal Form <i>(Submit an original, and a duplicate for fee processing)</i></li> <li><input checked="" type="checkbox"/> Specification, Claims, and Abstract Total Pages <u>33</u></li> <li><input checked="" type="checkbox"/> Drawings Total Sheets <u>6</u></li> <li>Oath or Declaration <i>(unexecuted)</i> Total Pages <u>4</u> <ol style="list-style-type: none"> <li><input type="checkbox"/> Newly executed (original or copy)</li> <li><input type="checkbox"/> Copy from prior application (37 CFR 1.63(d)) <i>(for continuation/divisional with Box 17 completed)</i>  <b>[Note Box 5 Below]</b>                (i) <input type="checkbox"/> <u>DELETION OF INVENTOR(S)</u>                Signed statement attached deleting inventor(s) named in the prior application, see 37 CFR 1.63(d)(2) and 1.33(b).</li> </ol> </li> <li><input type="checkbox"/> Incorporation by Reference <i>(usable if Box 4b is checked)</i> The entire disclosure of the prior application, from which a copy of the oath or declaration is supplied under Box 4b, is considered as being part of the disclosure of the accompanying application and is hereby incorporated by reference therein.</li> <li><input type="checkbox"/> Microfiche Computer Program <i>(Appendix)</i></li> <li><input type="checkbox"/> Nucleotide and/or Amino Acid Sequence Submission <i>(if applicable, all necessary)</i> <ol style="list-style-type: none"> <li><input type="checkbox"/> Computer Readable Copy</li> <li><input type="checkbox"/> Paper Copy (identical to computer copy)</li> <li><input type="checkbox"/> Statement verifying identity of above copies</li> </ol> </li> </ol>	<b>ACCOMPANYING APPLICATION PARTS</b> <ol style="list-style-type: none"> <li><input type="checkbox"/> Assignment Papers (cover sheet &amp; document(s))</li> <li><input type="checkbox"/> 37 CFR 3.73(b) Statement <i>(when there is an assignee)</i> <input type="checkbox"/> Power of Attorney by assignee</li> <li><input type="checkbox"/> English Translation Document <i>(if applicable)</i></li> <li><input type="checkbox"/> Information Disclosure Statement (IDS) PTO-1449 <input type="checkbox"/> Copies of IDS Citations</li> <li><input type="checkbox"/> Preliminary Amendment</li> <li><input checked="" type="checkbox"/> Return Receipt Postcard (MPEP 503) <i>(Should be specifically itemized)</i></li> <li><input type="checkbox"/> Small Entity Statement(s) <input type="checkbox"/> Statement filed in prior application Status still proper and desired</li> <li><input type="checkbox"/> Certified Copy of Priority Document(s)</li> <li><input checked="" type="checkbox"/> Other: <u>Four (4) unexecuted Declaration and Powers of Attorney</u></li> </ol>
17. If a <b>CONTINUING APPLICATION</b> , check appropriate box and supply the requisite information: <input type="checkbox"/> Continuation <input type="checkbox"/> Divisional <input type="checkbox"/> Continuation-in-part (CIP) of prior application No:	
18. CORRESPONDENCE ADDRESS:  W. Scott Petty, Esq. JONES & ASKEW, LLP 2400 Monarch Tower 3424 Peachtree Road, N.E. Atlanta, Georgia 30326	

By: <u><i>[Signature]</i></u>	Reg. No. <u>35,645</u>
Date: <u>June 15, 1999</u>	
Telephone: <u>404-949-2400</u>	
Facsimile: <u>404-949-2499</u>	

## SAFE SAVE METHOD FOR HTML DOCUMENTS

### FIELD OF THE INVENTION

The invention relates generally to saving a set of related files in a computer system, and more specifically to replacing an old version of an HTML document on a permanent storage medium with a new version of an HTML document in volatile memory. This allows either the replacement of the old HTML document or the restoration of the old HTML document should the replacement attempt be disrupted.

### BACKGROUND OF THE INVENTION

When a user modifies a document in a computer application, he or she usually wants to preserve those changes by replacing the old version of the document on a permanent storage medium, such as a hard drive or magnetic disk drive, with the new version of the document. Unfortunately, there exist many threats to the successful completion of this replacement. For example, the storage medium may run out of space, there could be a permission error, the computer might lose power, a network could go down, or the saving application could crash.

One way an application can save a document consisting of only one file is to save the file directly over the old document's file. If there is a failure at any point while the application is attempting to do this, however, the user ends up with a corrupted file that is a mix of bytes from the old file and bytes from the new file he or she was trying to save. If this happens, the user effectively loses both the old and the new versions of the document.

A way to reduce the possibility of this result is with a method, known as a "safe save", for saving a document consisting of only one file. This "safe

save” method can be used when a user has an old version of a document on a permanent storage medium that he wishes to replace with a new version of the document located in volatile memory, such as random access memory (RAM). The method is triggered by the user informing the application that the user wishes to save the document. In the “safe save” method, the application first saves the new version of the document under a temporary name on the permanent storage medium. If the application fails at any point during this step, the user still has the old version of the document. If the application completes this step, then both the old version and the new version of the file are on the permanent storage medium. The final step is to change the name of the file saved on the permanent storage medium under a temporary name to the original name of the file. This step is relatively fast, so the chance of corruption during this step is small. When this step is complete, the permanent storage medium contains the new version of the document.

Sometimes, a computer application modifies several related files at once. For example, the user may employ a word processing application to modify an HTML document. An HTML document typically has one main file in which other files are embedded. Thus, an HTML document may comprise many related files.

Suppose a user of a computer application wishes to replace old files on a permanent storage medium with new files in volatile memory that represent a newer version of the old files. This situation can arise when saving an HTML document having many related files. The application could attempt a variation on the “safe save” method wherein the application first saves the new files to the permanent storage medium under temporary names. The application could then change the name of each new file saved on the permanent storage medium under a temporary name to the final name for that file. As already explained, however, there could be many new files. The more there are, the longer this step will take and the greater is the possibility that something will interrupt this step. If a problem does interrupt this step, then the storage medium will likely contain a mixture of old and new files, and there is no way to tell which is which.

Clearly, the traditional “safe save” method does not scale to the problem of replacing one set of files on a permanent storage medium with another set of files in volatile memory. Thus, there is a need in the art for a mechanism for determining if a save attempt has been successful. There is also a need in the art for attempting to complete an interrupted save attempt

of multiple files, and, if unable to do so, restoring the original set of files the application intended to replace.

As already explained, an HTML document typically comprises many related files because an HTML document usually has one main file in which other files are embedded. Due to the prevalence of HTML documents, there is a need in the art for a “safe save” method that can be used for saving an HTML document. In other words, there is a need in the art for attempting to complete an interrupted attempt to replace an old HTML document with a new HTML document, and, if unable to do so, restoring the old HTML document for which the replacement was attempted.

## SUMMARY OF THE INVENTION

The present invention provides a “safe save” mechanism for completing a file save operation by replacing a set of old files, also described as original files, on a permanent storage medium with a set of new or replacement files in volatile memory. In response to an event leaving the saving application in control but unable to complete the primary steps of the replacement, known as a non-catastrophic failure, a backout can be attempted. A backout is a process for restoring the old set of files to the permanent storage medium. In contrast, a roll forward is a process for completing an interrupted replacement attempt. An event that causes the saving application to lose control or prevents completion of a backout or roll forward is known as a catastrophic failure. In response to a catastrophic failure, a roll forward can be attempted at a later time to finish the replacement of the old files with the new files. The roll forward attempt is accomplished by using a recovery file created during the primary steps of the replacement.

Generally described, the present invention provides a “safe save” method of replacing old files having old names on a permanent storage medium with new files in volatile memory. Some of these files may be opaque files, which are old files which, as a result of a save operation, have become new files without undergoing any changes, other than possibly name changes. A non-opaque file is any file that is not an opaque file, and clearly must be either an old file or a new file, but cannot be both. The non-opaque new files are transferred onto the permanent storage medium and assigned temporary names. An opaque recovery file and a primary recovery file are created. The opaque recovery file lists the old name and a temporary name for each opaque file. The primary recovery file lists the temporary name and a final name for each new file, including the opaque files since opaque files are also new files. After replacing the names of the opaque files with their temporary names, the names of the non-opaque old files are changed from their old names to backup names. Then, the names of the new files, including the opaque files, are changed from their temporary names to their final names. Optionally, the recovery files and the old files can be deleted. The foregoing primary steps of the “safe save” method result in the replacement of the old files with the new files on the permanent storage medium.

If a non-catastrophic failure occurs during the primary steps of the replacement, a backout is attempted to restore the old files to the permanent

storage medium. In a backout, each old file is checked to determine if it still has its old name as its filename. If an old file does not have its old name as its filename, the name of that file is changed to its old name. The old files may be processed as described in two batches: the opaque files and the non-opaque old files.

If a catastrophic failure occurs during the primary steps, a backout attempt, or a roll forward attempt, then a roll forward can be attempted at a later time to complete the replacement of old files with new files. In a roll forward, each opaque file is checked to determine if its filename has been changed from the old name to the temporary name specified for the opaque file in the opaque recovery file. If the opaque file's name has not been changed from the old name to the temporary name, then the opaque file's name is changed to the temporary name. Then, each new file is checked to determine if its filename has been changed from the temporary name to the final name specified for the new file in the primary recovery file. If the new file's name has not been changed from the temporary name to the final name, then the new file's name is changed to the final name. According to an aspect of the invention, an attempt is made to detect a catastrophic failure only in response to a user attempting to load the new files into volatile memory from the permanent storage medium.

The various aspects of the present invention may be more clearly understood and appreciated from a review of the following detailed description of the disclosed embodiments and by reference to the appended drawings and claims.

## BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a block diagram illustrating the operating environment for an exemplary embodiment of the present invention.

FIG. 2 is a flow chart illustrating the primary steps of an exemplary safe save method.

FIG. 3 is a chart illustrating a representative example of changes occurring within the contents of a hypothetical permanent storage medium in response to performing each of the primary steps of the exemplary safe save method shown in Fig. 2.

FIG. 4 is a flow chart illustrating the steps of an exemplary opaque backout process.

FIG. 5 is a flow chart illustrating the steps of an exemplary full backout process.

FIG. 6 is a flow chart illustrating the steps of an exemplary roll forward process.

## DETAILED DESCRIPTION OF THE EXEMPLARY EMBODIMENTS

When a user modifies one or more old files in a computer application, he usually does so in volatile memory. A user may modify a set of old files not only by changing the contents of individual old files, but also by adding or deleting entire files in order to form a set of new files. After modifying the old files, he usually wants to preserve those changes by replacing the old files on a permanent storage medium with the new files in volatile memory. The present invention includes a "safe save" method for doing this that has additional benefits.

In order to understand the primary steps of the "safe save" method, one must first be familiar with several definitions. An opaque file is an old file that has undergone no modification in becoming a new file, other than possibly undergoing a name change as a result of a save operation. Thus, an opaque file is both an old file and a new file. A non-opaque file is any file that is not an opaque file. Clearly, a non-opaque file is either an old file or a new file, but cannot be both.

The term permanent storage medium is used throughout this application to refer to a recording medium that retains the data recorded on it for long periods of time without power, such as some form of magnetic medium (e.g., floppy disk or tape), hard drive, or optical disk. The term volatile memory refers to memory, such as random access memory (RAM), that loses its data when power is disconnected from the system.

Generally, the present invention provides a "safe save" method of completing a file save operation by replacing old files having old names on a permanent storage medium with new files in volatile memory. The non-opaque new files are transferred onto the permanent storage medium and assigned temporary names. An opaque recovery file and a primary recovery file are then created. The opaque recovery file lists the old name and a temporary name for each opaque file. The primary recovery file lists the temporary name and a final name for each new file, including the opaque files. After replacing the names of the opaque files with their temporary names, the names of the non-opaque old files are changed from their old names to backup names. Then, the names of the new files are changed from their temporary names to their final names. Optionally, the recovery files

and the old files can be deleted. The foregoing primary steps of the "safe save" method results in the replacement of the old files with the new files on the permanent storage medium.

Unfortunately, many events can interfere with an attempt to replace old files with new files. For example, the storage medium may run out of space, there could be a permission error, the computer might lose power, a network could go down, or the saving application could crash. These events can be grouped into two categories: catastrophic failures and non-catastrophic failures.

The first category of events that might interfere with a file replacement attempt consists of non-catastrophic failures. A non-catastrophic failure is any event that leaves the program attempting the replacement in control but unable to complete the primary steps of the replacement. Non-catastrophic failures include running out of disk space, encountering a permissions error, and encountering a media failure.

If a non-catastrophic failure occurs, the replacement of old files with new files cannot be completed, by definition. If nothing else were done to the files on the permanent storage medium after a non-catastrophic failure, the user would be left with a mixture of old files and new files, and he would have trouble telling what files were old and what files were new. In many cases, a better situation for the user is to have the old files restored to the permanent storage medium.

The present invention provides a method for handling non-catastrophic failures during the primary steps of a replacement attempt during a file save operation. A backout is a process for restoring the old files to the permanent storage medium. Though a backout is attempted in response to a non-catastrophic failure, not all backout attempts will be successful. For example, if a non-catastrophic failure is generated by loss of access to the permanent storage medium, perhaps by it becoming unplugged, then the permanent storage medium cannot be accessed to do the backout.

The second category of events that might interfere with a replacement attempt consists of catastrophic failures. A catastrophic failure is defined as either of the following: 1) an event that causes the program attempting the replacement to lose control, or 2) an event that prevents completion of a backout or a roll forward. To clarify the definition of a catastrophic failure, a brief definition of a roll forward is given now, though the term will be explained in greater detail later. A roll forward is a process for completing a replacement attempt that has been interrupted by a



catastrophic failure, and the roll forward is accomplished using the recovery files created during the primary steps of the replacement. A catastrophic failure of the first type can occur if there is a power outage, the program attempting the replacement crashes, or the network goes down. An example of the second type of catastrophic failure is where a non-catastrophic failure is caused by loss of access to the permanent storage medium, thereby also preventing a successful backout.

The present invention also provides a method for handling catastrophic failures during an attempt to replace old files with new files. If a catastrophic failure occurs during a replacement attempt and is later detected, a roll forward is attempted to complete the replacement of old files with new files. As a catastrophic failure of the second type includes events that prevent completion of a roll forward, it should be understood that a failed roll forward attempt leads to another roll forward attempt, though preferably at a later time. In a roll forward attempt, the opaque recovery file is used to determine if any of the opaque files still have their old names. If they do, the names of those opaque files are changed to the temporary names specified for them in the opaque recovery file. Then, the primary recovery file is used to determine if any of the new files still have their temporary names. If they do, the names of those new files are changed to the final names specified for them in the primary recovery file.

The detailed description will elaborate on aspects of the present invention. First, an exemplary operating environment for this invention is described. Then, details of embodiments of the present invention in a HyperText Markup Language (HTML) environment are given. Referring now to the drawings, in which like numerals reference like elements throughout the several figures, aspects of the present invention and the exemplary operating environment will be described.

### **Exemplary Operating Environment**

FIG. 1 and the following discussion are intended to provide a brief and general description of a suitable computing environment **100** for an implementation of the present invention. The exemplary operating environment **100** includes a conventional personal computer system **120**, including a processing unit **121**, a system memory **122**, and a system bus **123** that couples the system memory **122** to the processing unit **121**. The system memory **122** includes read only memory (ROM) **124** and RAM **125**. A basic input/output system **126** (BIOS), containing the basic routines that

help to transfer information between elements within the personal computer system **120**, such as during start-up, is stored in ROM **124**.

The personal computer system **120** further includes a hard disk drive **127**, a magnetic disk drive **128**, e.g., to read from or write to a removable magnetic disk **129**, and an optical disk drive **130**, e.g., for reading a CD-ROM disk **131** or to read from or write to other optical media. The hard disk drive **127**, magnetic disk drive **128**, and optical disk drive **130** are connected to the system bus **123** by a hard disk drive interface **132**, a magnetic disk drive interface **133**, and an optical drive interface **134**, respectively. The drives and their associated computer-readable media provide nonvolatile storage for the personal computer system **120**. Although the description of computer-readable media above refers to a hard disk, a removable magnetic disk and a CD-ROM disk, it should be appreciated by those skilled in the art that other types of media that are readable by a computer system, such as magnetic cassettes, flash memory cards, digital video disks, Bernoulli cartridges, and the like, may also be used in the exemplary operating environment.

A user may enter commands and information into the personal computer **120** through conventional input devices, including a keyboard **140** and pointing device, such as a mouse **142**. The computer system **120** may include additional input devices (not shown), such as a microphone, joystick, game pad, satellite dish, scanner, or the like. These and other input devices are often connected to the processing unit **121** through a serial port interface **146** that is coupled to the system bus, but may be connected by other interfaces, such as a game port or a universal serial bus (USB). A monitor **147** or other type of display device is also connected to the system bus **123** via an interface, such as a video adapter **148**. In addition to the monitor, personal computer systems typically include other peripheral output devices (not shown), such as speakers or printers.

The personal computer system **120** may operate in a networked environment using logical connections to one or more remote computer systems, such as a remote computer system **149**. The remote computer system **149** may be a server, a router, a peer device or other common network node, and typically includes many or all of the elements described relative to the personal computer system **120**, although only a memory storage device **150** has been illustrated in FIG. 1. The logical connections depicted in FIG. 1 include a local area network (LAN) **151** and a wide area network (WAN) **152**. Such networking environments are

commonplace in offices, enterprise-wide computer networks, intranets and the Internet.

When used in a LAN networking environment, the personal computer system **120** is connected to the LAN **151** through a network interface **153**. When used in a WAN networking environment, the personal computer system **120** typically includes a modem **154** or other means for establishing communications over a WAN **152**, such as the Internet. The modem **154**, which may be internal or external, is connected to the system bus **123** via the serial port interface **146**. In a networked environment, program modules depicted relative to the personal computer system **120**, or portions thereof, may be stored in the remote memory storage device **150**. It will be appreciated that the network connections shown are exemplary and other means of establishing a communications link between the computer systems may be used. It will be further appreciated that the invention could equivalently be implemented on host or server computer systems other than personal computer systems, and could equivalently be transmitted to the host computer system by means other than a CD-ROM, for example, by way of the network connection interface **153**.

A number of program modules may be stored in the drives and RAM **125** of the computer system **120**. Program modules control how the computer system **120** functions and interacts with the user, with I/O devices or with other computers. Program modules include routines, operating system **135**, application programs modules **138**, data structures, browsers, and other software or firmware components. An exemplary embodiment of the present invention comprises one or more "Safe Save" program modules **137** stored in RAM **125** of the computer **100**. Alternatively, the program modules **137** may be implemented as part of the application programs modules **138** or the operating system **135**.

Those skilled in the art will appreciate that the invention may be practiced with other computer system configurations, including hand-held devices, multiprocessor systems, microprocessor-based or programmable consumer electronics, minicomputers, mainframe computers, and the like. The invention may also be practiced in distributed computing environments where tasks are performed by remote processing devices that are linked through a communications network. In a distributed computing environment, program modules may be located in both local and remote memory storage devices.

## Replacing Old Files With New Files In An HTML Environment

One example of a related group of files for which the present invention is applicable is an HTML document. An HTML document typically has one main file in which other files are embedded. Thus, an HTML document may comprise many related files.

In order to modify an HTML document, a user may employ a word processing application running on a computer, such as the computer system 120. To do so, he typically loads an HTML document into volatile memory from a permanent storage medium. He then makes changes to the document in volatile memory, possibly deleting some embedded files and adding different ones. A suitable word processing application for modifying HTML documents is Microsoft's "WORD" program, which is produced and distributed by Microsoft Corporation of Redmond, Washington.

In an exemplary embodiment, the word processing application itself implements a "safe save" method when the user indicates that he wishes to save a modified document, perhaps by choosing the Save command from a menu. The user is indicating through this command that he wishes to replace the old HTML document on the permanent storage medium with the new HTML document in volatile memory.

FIG. 2 is a flow chart illustrating the primary steps of an exemplary "safe save" method 200. The primary steps are performed only in response to the user saving a modified document. If the user is saving an entirely new document, the files comprising that document are saved on the permanent storage medium, and no special handling is required.

At the "START" step 205, an old HTML document is maintained on a permanent storage medium. The old HTML document comprises one or more old files, alternatively described as original files. One of the old files is an old main file, or the primary original file, in which the remaining old files are embedded. Each of the old files has an old or original name, which is the filename under which those files were previously saved.

In volatile memory is a new HTML document that is a modified version of the old HTML document. The new HTML document comprises one or more new files, alternatively described as replacement files. One of the new files is a new main file, or the primary replacement file, in which the remaining new files are embedded. Each of the new files has a final name,

which is the name chosen by either the user or the saving application to be the filename of the new file once the old HTML document is replaced by the new HTML document.

Some of the old files are opaque files, which are old files that have undergone no modification in becoming new files, other than possibly undergoing a name change, as a result of a save operation. Neither the old main file nor the new main file is treated as an opaque file, even if they are identical.

FIG. 3 is an example illustrating the contents of a hypothetical permanent storage medium after each of the primary steps **200** of FIG. 2. Each column is labeled with one of the FIG. 2 steps. A column shows the file names of any files located on the permanent storage medium after the step labeling that column is performed. Each row of the chart corresponds to one file and shows the name changes that file undergoes as the primary steps **200** of FIG. 2 are performed. For example, the column labeled “Step 205” shows the old files and the file names under which they are stored on the permanent storage medium. Book1.htm is the old main file. Image003.png, image004.jpg, image005.png, and image006.gif are opaque files.

In step **210**, the non-opaque new files are transferred onto the permanent storage medium. At that time, the non-opaque new files are also assigned temporary names. The temporary name of each new file is preferably created by replacing the last character of the final name of the new file with the “\_” character. In FIG. 3, the column labeled “Step 210” shows the results of this step for the example.

After assigning temporary names to the new files, step **215** is executed. In step **215**, an opaque recovery file is created. The opaque recovery file typically lists the old name and a temporary name for each old file. Again, the temporary name of each opaque file is preferably created the same way the temporary name of each of the other new files is created, which is by replacing the last character of the final name of the opaque file with the “\_” character. The general format of a recovery file will be described below with respect to Table 1.

A recovery file can comprise multiple records, appended one after another, each of which specifies an operation to perform on a file, a “source filename” for that file, and a “destination filename” for that file. A 0 can specify a copy operation, a 1 can specify a rename operation, and a 2 can specify a delete operation. If a rename operation is specified in a record, then the record indicates that the name of the file whose name is “source

filename” is to be changed to “destination filename.” Preferably, only the rename operation is used for records of the opaque and primary recovery files. A representative format of a record in an exemplary recovery file is shown in Table 1 and an example of a recovery file record is shown in Table

5 2.

**Table 1**  
**Exemplary Recovery File**

10	Byte Offset	Contents
	0	Operation to perform
	1,..., 4	Length of source filename
	5,..., N	Source filename
	N+1,..., N+4	Length of destination filename
	N+5,...,N+M	Destination filename

**Table 2**  
**Recovery File Record**

(1)(15)”c:\Image003.png”(15)”c:\Image001.pn\_”

A recovery file also preferably has a begin file marker (also called a TOF marker, for Top Of File) and an end file marker (also called an EOF marker, for End Of File). The TOF and EOF markers designate the start and end points of the recovery file. They are also used to verify that the file is intact.

In Table 1, the Byte Offset indicates how many bytes from the beginning of the record one must move to find the information located in the adjacent Contents column. Thus, the first byte of information (0 byte offset) specifies the operation to perform on the file to which that record pertains. The next four bytes of the record are used to represent a number, which is the length in bytes of the source filename. Each character can be represented with one byte. The range of the fifth byte offset to the Nth byte offset defines the source filename. The next four bytes of the record are used to represent a number, which is the length in bytes of the destination filename. The range of the N+5 byte offset to the N+M byte offset defines the destination filename.

For the example of a recovery file record in Table 2, the parentheses indicate numbers, and the quotes indicate character strings. This record specifies a source filename “c:\Image003.png” which is fifteen characters long. The record also specifies a destination filename “c:\Image001.pn\_” which is fifteen characters long. As a rename operation is specified, the recovery file record indicates that the file whose name is currently “c:\Image003.png” is going to experience or has experienced a name change to “c:\Image001.pn\_”.

Returning to step 215, an opaque recovery file is created. The opaque recovery file preferably has the format for a recovery file shown in Table 1. The opaque recovery file can include one record for each opaque file. The source filename of an opaque file’s record is the old name of the opaque file, and the destination filename of an opaque file’s record is a temporary name for that opaque file. The temporary name can be created as already explained. Thus, the opaque recovery file lists the old name and a temporary name for each old file. The opaque recovery file can be stored on the permanent storage medium and assigned a name formed by taking the filename of the old main file and changing the extension to “.rc\_”.

In the example of FIG. 3, Book1.rc\_ is the opaque recovery file. Table 3 shows the information contained in the records of Book1.rc\_ for the example of FIG. 3.

**Table 3**

Rename image003.png → image001.pn_
Rename image004.jpg → image002.jp_
Rename image005.png → image003.pn_
Rename image006.gif → image004.gi_

In step 220, a primary recovery file is created. The primary recovery file can list the temporary name and the final name for each new file, including the opaque files. The primary recovery file’s format is preferably the same as the general format described above with respect to Table 1, which also defines the general format for the opaque recovery file. The primary recovery file typically has one record for each new file. The source filename of a new file’s record is the temporary name of the new file. The destination filename of the new file’s record is the final name determined by the user or the program to be the filename of the new file once the

replacement of the old HTML document by the new HTML document is complete on the permanent storage medium. The primary recovery file can be stored on the permanent storage medium and assigned a name formed by taking the filename of the old main file and changing the extension to “.rcv”.

In the example of FIG. 3, Book1.rcv is the primary recovery file. Table 4 shows the information contained in the records of Book1.rcv for the example of FIG. 3.

**Table 4**

Rename Book1.ht_ → Book1.htm
Rename editdata.ms_ → edidata.mso
Rename stylesheet.cs_ → stylesheet.css
Rename tabstrip.ht_ → tapstrip.htm
Rename sheet001.ht_ → sheet001.htm
Rename image001.pn_ → image001.png
Rename image002.jp_ → image002.jpg
Rename image003.pn_ → image003.png
Rename image004.gi_ → image004.gif
Rename sheet002.ht_ → sheet002.htm
Rename sheet003.ht_ → sheet003.htm
Rename filelist.xm_ → filelist.xml

After creating the primary recovery file, step **225** is performed. In step **225**, the names of the opaque files are changed from their old names to their temporary names. Table 5 lists the name changes that are performed by this step in the example of FIG. 3.

**Table 5**

Rename image003.png → image001.pn_
Rename image004.jpg → image002.jp_
Rename image005.png → image003.pn_
Rename image006.gif → image004.gi_

After the names of the opaque files are changed from their old names to their temporary names, the opaque recovery file is deleted in step **230**.

In step **235**, the name of the old main file is changed from its old name to a backup name. The backup name for any old file, including the



old main file, is preferably created by replacing the last character of the old file's name with the “~” character. In step **240**, the name of the new main file is changed from its temporary name to its final name.

In step **245**, the names of the non-opaque old files are changed from their old names to their backup names. Even though the old main file may be a non-opaque old file, the old main file is excluded from this step because the name of the old main file was already changed from its old name to its backup name in step **235**. This step can be performed by attempting to change the names of all the old files, including the opaque files, from their old names to their backup names. However, the opaque files will not be found under their old names at this step because the opaque files already had their names changed in step **225** from their old names to their temporary names. Hence, a “file not found” error may be generated, which can be ignored so that the name change is not done for the opaque files in this step. Table 6 lists the name changes that are attempted in this step for the example of FIG. 3.

**Table 6**

Rename editdata.mso → edidata.ms~
Rename stylesheet.css → stylesheet.cs~
Rename tabstrip.htm → tapstrip.ht~
Rename sheet001.htm → sheet001.ht~
Rename image001.png → image001.pn~
Rename image002.gif → image002.gi~
Rename image003.png → image003.pn~
Rename image004.jpg → image004.jp~
Rename image005.png → image005.pn~
Rename image006.gif → image006.gi~
Rename sheet002.htm → sheet002.ht~
Rename sheet003.htm → sheet003.ht~
Rename filelist.xml → filelist.xm~

In step **250**, the names of the new files are changed from their temporary names to their final names. Although the new main file is a new file, it is excluded from this step because the new main file already had its name changed from its temporary name to its final name in step **240**. Table

7 lists the name changes that are performed by this step for the example of FIG. 3.

**Table 7**

Rename editdata.ms\_ → editdata.mso  
 Rename stylesheet.cs\_ → stylesheet.css  
 Rename tabstrip.ht\_ → tapstrip.htm  
 Rename sheet001.ht\_ → sheet001.htm  
 Rename image001.pn\_ → image001.png  
 Rename image002.jp\_ → image002.jpg  
 Rename image003.pn\_ → image003.png  
 Rename image004.gi\_ → image004.gif  
 Rename sheet002.ht\_ → sheet002.htm  
 Rename sheet003.ht\_ → sheet003.htm  
 Rename filelist.xm\_ → filelist.xml

In step **255**, the backup files are deleted from the permanent storage medium. The backup files are all those with backup names on the permanent storage medium. These are the non-opaque old files. Table 8 lists the files that are deleted by this step for the example of FIG. 3.

**Table 8**

Delete Book1.ht~  
 Delete editdata.ms~  
 Delete stylesheet.cs~  
 Delete tabstrip.ht~  
 Delete sheet001.ht~  
 Delete image001.pn~  
 Delete image002.gi~  
 Delete image003.pn~  
 Delete image004.jp~  
 Delete image005.pn~  
 Delete image006.gi~  
 Delete sheet002.ht~  
 Delete sheet003.ht~  
 Delete filelist.xm~

In step **260**, the primary recovery file is deleted. Then the primary step **200** terminates at the "END" step **270**.

If the method **200** is performed without a catastrophic or non-catastrophic failure occurring, the old files will have been successfully replaced on the permanent storage medium with the new files. If, however, a catastrophic or non-catastrophic failure occurs during this "safe save" method, then alternative procedures can be completed in an effort to recover from the catastrophic or non-catastrophic failure. These procedures, explained in FIGS. 4-6 and the accompanying text, will be discussed shortly.

Though the importance of steps **235** and **240** will be greater appreciated after a review of the description of FIG. 6, a brief explanation is given now. FIG. 6 is a flowchart of an exemplary roll forward process **600** that is performed after a catastrophic failure during the exemplary primary steps **200** of the "safe save" method. In one embodiment of the present invention, a roll forward process **600** is initiated when the user attempts to access a new HTML document. This access is typically attempted by trying to load the new HTML document into volatile memory from the permanent storage medium at some point after an unsuccessful attempt to replace the old HTML document with the new HTML document on the permanent storage medium. To attempt to load the new HTML document into volatile memory, however, the user has to locate the new main file. The user will only be able to find the new main file if it exists on the permanent storage medium under the filename the user expects it to have--i.e., its final name.

In order to initiate the roll forward process **600** in one embodiment of the invention, therefore, a copy of the new main file must be on the permanent storage medium with its final name as the filename. Steps **235** and **240**, shown in Fig. 2, represent quick operations, so there will always be a copy of the new main file to initiate the roll forward process **600** should a catastrophic failure occur during the subsequent step **245** of changing the names of the remaining non-opaque old files from their old names to their backup names. In this manner, a copy of the new main file is also available to initiate the roll forward process **600** should a catastrophic failure occur when changing the names of the remaining new files to their final names in step **250**.

If steps **235** and **240** were not performed, there might not be a copy of the new main file on the permanent storage medium allowing the user to initiate the roll forward process **600**. An example illustrating this will now be presented. Assume that, instead of performing steps **235** and

240, the names of all the non-opaque old files (including the old main file) are changed to their backup names in step 245. Assume further that in the subsequent step 250, the names of all the new files (including the new main file) are changed to their final names. If a catastrophic failure occurs during step 245 after the name of the old main file has been changed to the backup name, then no copy of the new main file is available on the permanent storage medium to allow the user to initiate the roll forward process 600.

FIG. 4 is a flow chart of the steps for an exemplary opaque backout process 400. It is performed in response to a non-catastrophic error that occurs through step 235 of the exemplary "safe save" method 200 illustrated in FIG. 2. The purpose of the opaque backout process 400 is to restore the old files to their old names on the permanent storage medium since the replacement cannot be completed due to the non-catastrophic error. As the only old files whose names are changed through step 235 are the opaque files, the opaque backout process 400 proceeds by insuring that all the opaque files are restored to their old names.

The opaque backout process 400 starts at step 410 and proceeds to step 420. Step 420 effectively begins a loop for processing each of the opaque files. In step 420, an inquiry is conducted to determine if there is another opaque file to process. If there is not, then the opaque backout process 400 ends at step 430.

If there is another opaque file to process, then that opaque file is selected as the current file in step 440. In step 450, an inquiry is conducted to determine whether the temporary name associated with the current file can be found on the permanent storage medium. If the temporary name is not found, then the opaque file that is the current file has never had its name changed from its old name to its temporary name, and control passes back to step 420 to determine if there is another opaque file to process. If the temporary name is found, however, then step 460 is performed. In step 460, an attempt is made to change the name of the opaque file which is the current file from its temporary name back to its old name.

In step 470, an inquiry is conducted to determine whether this name change was successful. If the name change was not successful, then a catastrophic failure has occurred and the process ends at step 430. When the user attempts to open the document at a later time, only a roll forward can be attempted. An exemplary roll forward process 600 is illustrated in FIG. 6 and shall be discussed. On the other hand, if a determination is made in step

470 that the name change was successful, then control passes back to step 420 to determine if there is another opaque file to process.

FIG. 5 is a flow chart of the steps for an exemplary full backout process 500. It is performed in response to a non-catastrophic error occurring during steps 240, 245, or 250 of the primary steps 200 of the exemplary "safe save" method illustrated in FIG. 2. The purpose of the full backout process 500 is to restore the old files to their old names on the permanent storage medium. If a non-catastrophic error occurs during steps 240, 245 or 250, then both opaque and non-opaque old files have undergone name changes. Hence, the full backout process 500 proceeds by first restoring all the non-opaque old files to their old names on the permanent storage medium, and then restoring all the opaque files to their old names.

The full backout process 500 starts at step 510 and proceeds to step 520. Step 520 effectively begins a loop for processing all of the non-opaque old files. In step 520, an inquiry is conducted to determine if there is another non-opaque old file to process. If there is another non-opaque old file to process, then that non-opaque old file is selected as the current file in step 530.

In step 540, an inquiry is conducted to determine whether the backup name associated with the current file can be found on the permanent storage medium. If the backup name associated with the current file is not found, then the non-opaque old file has not yet had its name changed from its old name to its backup name, and control passes back to step 520 to determine if there is another non-opaque old file to process. If, on the other hand, the backup name associated with the current file is found on the permanent storage medium, then an attempt is made to change the name of the non-opaque old file that is the current file from its backup name to its old name in step 550.

In step 560, an inquiry is made to determine whether the name change was successful. If the name change was not successful, then a catastrophic failure has occurred and the process ends at step 580. When the user attempts to open the document at a later time, only a roll forward can be attempted. An exemplary roll forward process 600 is illustrated in FIG. 6 and will be discussed. If, on the other hand, a determination is made in step 560 that the name change was successful, then control returns to step 520 to determine if there is another non-opaque old file to process.

In step 520, if there is not another non-opaque old file to process, then all the non-opaque old files have been restored to their old

names on the permanent storage medium. At that point, only the opaque files need to be restored to their old names on the permanent storage medium. So, the opaque backout process **400** is then performed before the process ends at step **580**.

5           FIG. 6 is a flow chart of the steps for an exemplary roll forward process **600** for handling a catastrophic failure during any of the processes of the exemplary “safe save” method illustrated in FIGS. 2, 4, or 5. In one embodiment in the present invention, the roll forward process **600** is initiated in step **605** when the user attempts to access an HTML document by  
10       loading it into volatile memory from the permanent storage medium.

          The HTML document the user is attempting to load into volatile memory in step **605** may represent the new HTML document based on an unsuccessful attempt to replace an old HTML with a new HTML document. In this case, a catastrophic failure has disrupted the attempt to replace the old  
15       HTML document. Alternatively, there may be no problem loading the HTML document into volatile memory because there was no catastrophic failure. Thus, a determination must be made concerning whether there has been a catastrophic failure that warrants further action before loading the requested document into volatile memory.

20           In one embodiment of the present invention, this determination is made based upon whether the opaque and primary recovery files are available to support file recovery operations. If the opaque recovery file is available, then a catastrophic failure occurred before the names of all the opaque files were changed from their old names to their temporary names  
25       since the opaque recovery file is deleted after that step is completed. Thus, if the opaque recovery file is found, the roll forward process **600** insures that the names of all the opaque files are changed to their temporary names. If the primary recovery file is found, then there was a catastrophic failure before the names of all the new files were changed from their temporary  
30       names to their final names. Thus, if the primary recovery file is found, the roll forward process **600** insures that the names of all the new files have been changed to their final names.

          After the user attempts to load an HTML document into volatile memory in step **605**, an inquiry is conducted in step **610** to determine  
35       whether an opaque recovery file can be found in a predetermined location, which is the permanent storage medium in one embodiment of the present invention. If the opaque recovery file is not found, control proceeds to step

**655** to attempt to determine if there is a primary recovery file to process. If the opaque recovery file is found, then step **615** is performed.

In step **615**, an inquiry is conducted to determine whether the opaque recovery file is intact. The opaque recovery file is considered intact if it has valid TOF and EOF markers. If the opaque recovery file is not intact, the opaque recovery file is deleted in step **620**, before proceeding to step **655** to determine whether there is a primary recovery file to be processed. If the opaque recovery file is intact, then step **625** is performed.

Step **625** effectively begins a loop to process all the records of the opaque recovery file. An inquiry is conducted to determine whether there is another record of the opaque recovery file to process. If there is not another record, then the loop is complete. The opaque recovery file is deleted in step **620** before proceeding to step **655** to determine whether there is a primary recovery file to process. If there is another record of the opaque recovery file to process in step **625**, then that record is selected in step **630**.

In step **635**, an inquiry is conducted to determine whether a valid operation is specified in the operation byte of the selected record. An operation is considered valid if the operation byte is set to a 0, 1, or 2. If a valid operation is not specified, then it is assumed that the opaque recovery file is not intact. This file is deleted in step **620** before proceeding to step **655** to determine whether there is a primary recovery file to process. If a valid operation is specified in step **635**, then step **640** is executed.

In step **640**, an inquiry is conducted to determine whether a file having as its filename the old name specified in the selected record is located on the permanent storage medium. If a file having the old name as its filename is not located on the permanent storage medium, then control passes back to step **625** to determine if there is another record of the opaque recovery file to process. If a file having the old name as its filename is located on the permanent storage medium, then the opaque file to which the selected record refers has not yet had its name changed from its old name to its temporary name. In that case, an attempt is made to change the name of the opaque file from its old name to its temporary name in step **645**.

In step **650**, an inquiry is conducted to determine whether the name change was successfully performed. If the name change was not successfully performed, then this is a catastrophic failure and the process ends at step **699**. In this case, the roll forward process **600** will be attempted once again if the user attempts to load the new HTML document

into volatile memory at a later time. If the name change was successfully performed, then control passes back to step **625** to determine whether there is another record of the opaque recovery file to process.

In step **655**, an inquiry is conducted to determine whether there is a primary recovery file to process. This decision is made by looking for the primary recovery file in a predetermined location, which is the permanent storage medium in one embodiment of the present invention. If the primary recovery file is not found, the process ends at step **699**. If the primary recovery file is found, then step **660** is executed.

In step **660**, an inquiry is made to determine whether the primary recovery file is intact. The primary recovery file is considered intact if it has valid TOF and EOF markers. If the primary recovery file is not intact, then the primary recovery file is deleted in step **665** before the process ends at step **699**. If the primary recovery file is intact in step **660**, then step **670** is performed.

Step **670** effectively begins a loop to process all the records of the primary recovery file. An inquiry is conducted to determine whether there is another record of the primary recovery file to process. If there is not another record, then the loop is complete, and the primary recovery file is deleted in step **665** before the process ends at step **699**. If there is another record of the primary recovery file to process in step **670**, then that record is selected in step **675**.

In step **680**, an inquiry is conducted to determine whether a valid operation is specified in the operation byte of the selected record. An operation is considered to be valid if the operation byte is set to 0, 1 or 2. If a valid operation is not specified, then it is assumed that the primary recovery file is not intact. This file is deleted in step **665** before the process ends at step **699**. If a valid operation is specified in the selected record in step **680**, then step **685** is executed.

In step **685**, an inquiry is conducted to determine whether a file having as its filename the temporary name specified in the selected record is located on the permanent storage medium. If a file having the temporary name as its filename is not located on the permanent storage medium, then control passes back to step **670** to determine whether there is another record of the primary recovery file to process. If a file having the temporary name as its filename is located on the permanent storage medium, then the new file has not yet had its name changed from its temporary name to its final name.



In that case, an attempt is made to change the name of the new file from its temporary name to its final name in step 690.

In step 695, an inquiry is made to determine whether this name change was successfully performed. If the name change was not successfully performed, then a catastrophic failure has occurred and the process ends at step 699. In that case, the roll forward process 600 will be attempted again if the user attempts to load the new HTML document into volatile memory at a later time. If the name change was successfully performed, however, control passes back to step 670 to determine whether there is another record of the primary recovery file to process.

The invention may conveniently be implemented in one or more program modules, such as the Safe Save Program Module 137, that are based upon the methods described in the detailed description. No particular programming language has been described for carrying out the various procedures described above because it is considered that the operations, steps, and procedures described above and illustrated in the accompanying drawings are sufficiently disclosed to permit one of ordinary skill in the art to practice an exemplary embodiment. Moreover, there are many computers and operating systems which may be used in practicing an exemplary embodiment and therefore no detailed computer program could be provided which would be applicable to all of these many different systems. Each user of a particular computer will be aware of the language and tools which are most useful for that user's needs and purposes.

One skilled in the art should recognize that it is possible to implement the present invention without distinguishing between opaque and non-opaque files. In the primary steps of the "safe save" method, this would require saving a copy of all new files in volatile memory to the permanent storage medium instead of doing this only for the non-opaque new files. That way, there is no need to create an opaque recovery file, nor is there a need to change to change the names of the opaque files from their old names to their temporary names. Then, the names of all the old files are changed from their old names to their backup names. Finally, the names of the new files are changed from their temporary names to their final names. The backout and roll forward processes would be modified accordingly.

One skilled in the art should also recognize that the recovery files can be implemented in different ways. One recovery file can list all the name change records, or multiple files can be used to hold the name change

records. Similarly, the deletion of recovery files and backup files can be performed in different ways.

One skilled in the art should also recognize that there are different ways to initiate the recovery process after a catastrophic failure.

5 For example, instead of the process being initiated by the user attempting to load a new version of a document after a catastrophic failure, a search for recovery files could automatically be conducted upon loading of the operating system or the applicable application software. The recovery process could also be initiated by a specific command from the user.

10 The method of the present invention could also be used to replace files on one permanent storage medium with files from a different permanent storage medium. Also, different kinds of connected files other than HTML files will occur to one skilled in the art.

15 Other alternative embodiments will become apparent to those skilled in the art to which an exemplary embodiment pertains without departing from its spirit and scope. Accordingly, the scope of an exemplary embodiment is defined by the appended claims rather than the foregoing description.

## CLAIMS

What is claimed is:

5           1.     A computer-implemented method for completing a file save operation by replacing original files with replacement files, comprising the steps:

                  initiating the file save operation in response to creating the replacement files;

10                assigning a temporary name to each replacement file;

                  saving the replacement files, each having a corresponding temporary name, from a volatile memory onto a permanent storage medium;

                  creating a recovery file comprising information useful for replacing the original files with the replacement files in the event of a catastrophic failure that interrupts the file save operation; and

15                replacing the temporary name of each replacement file with a corresponding final name to complete the file save operation.

20           2.     The method of claim 1, further comprising the step of changing an original name assigned to each original file to a backup name to support a restoration of the assignment of the original name to the original file in the event of a non-catastrophic failure preventing completion of the file save operation.

25           3.     The method of claim 2, further comprising the steps of deleting the recovery file and deleting each original file having a corresponding backup name in response to successfully completing the file save operation.

4. The method of claim 1, further comprising the following steps:  
determining whether a non-catastrophic failure of the file save  
operation has occurred;

if a non-catastrophic failure is detected, then

5 determining if a present file name of one of the original  
files is identical to the corresponding original name;

if the present file name is not identical to the original  
name, then replacing the present filename of the original file with the  
original name.

10 5. The method of claim 1, wherein the information of the recovery  
file comprises a list defining the temporary name and the final name for each  
replacement file, and wherein the method further comprises the steps:

15 determining whether a catastrophic failure of the file save  
operation has occurred;

if a catastrophic failure is detected, then

20 determining if a present file name of one of the  
replacement files is identical to the corresponding final name listed in the  
recovery file;

if the present file name is not identical to the final name,  
then replacing the present filename of the replacement file with the final  
name listed in the recovery file.

25 6. The method of claim 5, wherein the step of determining whether  
a catastrophic failure has occurred is completed in response to an attempt to  
load the replacement files into the volatile memory from the permanent  
storage medium.

30 7. The method of claim 6, wherein a presence of the recovery file  
on the permanent storage medium indicates that the catastrophic failure has  
occurred.

8. A computer-implemented method for completing a file save operation by replacing old files with new files, the old files assigned old names and comprising opaque files and non-opaque old files, the new file assigned final names and comprising the opaque files and non-opaque new files, wherein each opaque file represents static content of one of the old files and one of the new files, the method comprising the steps:

initiating the file save operation by creating the new files;

assigning a temporary name to the opaque files and to the non-opaque files for each new file;

saving the non-opaque new files onto the permanent storage medium, each of the non-opaque new files having a corresponding temporary name;

creating an opaque recovery file comprising information listing the old name and the temporary name for each opaque file, the information of the opaque recovery file useful for supporting replacement of the old files with the new files in the event of a catastrophic failure of the file save operation;

creating a primary recovery file comprising information listing the temporary name and the final name of the opaque files and the non-opaque new files for each new file, the information of the primary recovery file useful for supporting replacement of the old files with the new files in the event of a catastrophic failure of the file save operation;

replacing the old names of the opaque files with the temporary names;

replacing the old names of the non-opaque old files with backup names to support a restoration of the assignment of the old name to the non-opaque old file in the event of a non-catastrophic failure preventing completion of the file save operation; and

replacing the temporary names of the opaque files and the non-opaque new files with the final names for the new files to complete the file save operation.

9. The method of claim 8, further comprising the steps of deleting the opaque recovery file and deleting the primary recovery file in response to successfully completing the file save operation.

10. The method of claim 8, further comprising the step of deleting the non-opaque old files in response to replacing the temporary names of the opaque files and non-opaque new files with the final names for the new files.

5 11. The method of claim 8, further comprising the following steps:  
determining whether a non-catastrophic failure of the file save operation has occurred;

if a non-catastrophic failure is detected, then

10 determining whether a present file name for one of the opaque files is not identical to a corresponding old name;

if the present file name for the opaque file is not identical to the corresponding old name, then changing the present file name of the opaque file to the corresponding old name;

15 determining whether a present file name for a non-opaque old file is not is not identical to a corresponding old name;

if the present file name for the non-opaque old file is not identical to the corresponding old name, then changing the present file name of the non-opaque old file to the corresponding old name.

20 12. The method of claim 8, further comprising the following steps:  
determining whether a catastrophic failure of the file save operation has occurred;

if a catastrophic failure of the file save operation is detected, then

25 for each opaque file:

if the present file name for the opaque file is not identical to the temporary name listed in the opaque recovery file for the opaque file, then changing the present file name of the opaque file to the temporary name, and

30 for each new file:

if the present file name for the new file is not identical to the final name listed in the primary recovery file, then changing the present file name of the new file to the final name.

5

10

15. A computer-implemented method for recovering from an interrupted file save operation by using a primary recovery file to continue an attempt to replace old files with new files, the primary recovery file comprising information listing a temporary name and a final name for each new file, comprising the steps:

determining if a catastrophic failure of the file save operation has occurred during an attempt to replace the old files with the new files;

if a catastrophic failure of the file save operation has occurred, then determining whether the temporary name for one of the new files has been changed to a present file name identical to the final name listed in the primary recovery file for the new file,

if the present file name of the new file is not identical to the final name, then changing the present file name of the new file to the final name.

16. The method of claim 15, wherein the old files comprise opaque files and non-opaque old files and the new files comprise the opaque files and non-opaque new files, each opaque file representing static content of one of the old files and the new files, and wherein an opaque recovery file comprises information listing an old name and a temporary name for each opaque file, the method further comprising the steps:

if a catastrophic failure has occurred during an attempt to replace the old files with the new files,

then determining whether the old name for one of the opaque files has been changed to a present file name of the temporary name listed in the opaque recovery file for the opaque file,

if the present file name for the opaque file is not identical to the temporary name, then changing the present file name of the opaque file to the temporary name.

17. The method of claim 16, wherein the step of determining whether a catastrophic failure has occurred is completed in response to an attempt to load the new files into volatile memory from a permanent storage medium.



18. The method of claim 16, wherein a presence on a permanent storage medium of either the opaque recovery file or the primary recovery file at the time an attempt is made to load the new files into volatile memory from the permanent storage medium indicates that a catastrophic failure of the file save operation has occurred.

19. A computer-readable medium having computer-executable instructions for performing the method recited in claim 15.

20. The method of Claim 15, wherein the old files comprise a primary HTML and zero or more embedded files.

6651 P.010 " 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288 289 290 291 292 293 294 295 296 297 298 299 300 301 302 303 304 305 306 307 308 309 310 311 312 313 314 315 316 317 318 319 320 321 322 323 324 325 326 327 328 329 330 331 332 333 334 335 336 337 338 339 340 341 342 343 344 345 346 347 348 349 350 351 352 353 354 355 356 357 358 359 360 361 362 363 364 365 366 367 368 369 370 371 372 373 374 375 376 377 378 379 380 381 382 383 384 385 386 387 388 389 390 391 392 393 394 395 396 397 398 399 400 401 402 403 404 405 406 407 408 409 410 411 412 413 414 415 416 417 418 419 420 421 422 423 424 425 426 427 428 429 430 431 432 433 434 435 436 437 438 439 440 441 442 443 444 445 446 447 448 449 450 451 452 453 454 455 456 457 458 459 460 461 462 463 464 465 466 467 468 469 470 471 472 473 474 475 476 477 478 479 480 481 482 483 484 485 486 487 488 489 490 491 492 493 494 495 496 497 498 499 500 501 502 503 504 505 506 507 508 509 510 511 512 513 514 515 516 517 518 519 520 521 522 523 524 525 526 527 528 529 530 531 532 533 534 535 536 537 538 539 540 541 542 543 544 545 546 547 548 549 550 551 552 553 554 555 556 557 558 559 560 561 562 563 564 565 566 567 568 569 570 571 572 573 574 575 576 577 578 579 580 581 582 583 584 585 586 587 588 589 590 591 592 593 594 595 596 597 598 599 600 601 602 603 604 605 606 607 608 609 610 611 612 613 614 615 616 617 618 619 620 621 622 623 624 625 626 627 628 629 630 631 632 633 634 635 636 637 638 639 640 641 642 643 644 645 646 647 648 649 650 651 652 653 654 655 656 657 658 659 660 661 662 663 664 665 666 667 668 669 670 671 672 673 674 675 676 677 678 679 680 681 682 683 684 685 686 687 688 689 690 691 692 693 694 695 696 697 698 699 700 701 702 703 704 705 706 707 708 709 710 711 712 713 714 715 716 717 718 719 720 721 722 723 724 725 726 727 728 729 730 731 732 733 734 735 736 737 738 739 740 741 742 743 744 745 746 747 748 749 750 751 752 753 754 755 756 757 758 759 760 761 762 763 764 765 766 767 768 769 770 771 772 773 774 775 776 777 778 779 780 781 782 783 784 785 786 787 788 789 790 791 792 793 794 795 796 797 798 799 800 801 802 803 804 805 806 807 808 809 810 811 812 813 814 815 816 817 818 819 820 821 822 823 824 825 826 827 828 829 830 831 832 833 834 835 836 837 838 839 840 841 842 843 844 845 846 847 848 849 850 851 852 853 854 855 856 857 858 859 860 861 862 863 864 865 866 867 868 869 870 871 872 873 874 875 876 877 878 879 880 881 882 883 884 885 886 887 888 889 890 891 892 893 894 895 896 897 898 899 900 901 902 903 904 905 906 907 908 909 910 911 912 913 914 915 916 917 918 919 920 921 922 923 924 925 926 927 928 929 930 931 932 933 934 935 936 937 938 939 940 941 942 943 944 945 946 947 948 949 950 951 952 953 954 955 956 957 958 959 960 961 962 963 964 965 966 967 968 969 970 971 972 973 974 975 976 977 978 979 980 981 982 983 984 985 986 987 988 989 990 991 992 993 994 995 996 997 998 999 1000 1001 1002 1003 1004 1005 1006 1007 1008 1009 1010 1011 1012 1013 1014 1015 1016 1017 1018 1019 1020 1021 1022 1023 1024 1025 1026 1027 1028 1029 1030 1031 1032 1033 1034 1035 1036 1037 1038 1039 1040 1041 1042 1043 1044 1045 1046 1047 1048 1049 1050 1051 1052 1053 1054 1055 1056 1057 1058 1059 1060 1061 1062 1063 1064 1065 1066 1067 1068 1069 1070 1071 1072 1073 1074 1075 1076 1077 1078 1079 1080 1081 1082 1083 1084 1085 1086 1087 1088 1089 1090 1091 1092 1093 1094 1095 1096 1097 1098 1099 1100 1101 1102 1103 1104 1105 1106 1107 1108 1109 1110 1111 1112 1113 1114 1115 1116 1117 1118 1119 1120 1121 1122 1123 1124 1125 1126 1127 1128 1129 1130 1131 1132 1133 1134 1135 1136 1137 1138 1139 1140 1141 1142 1143 1144 1145 1146 1147 1148 1149 1150 1151 1152 1153 1154 1155 1156 1157 1158 1159 1160 1161 1162 1163 1164 1165 1166 1167 1168 1169 1170 1171 1172 1173 1174 1175 1176 1177 1178 1179 1180 1181 1182 1183 1184 1185 1186 1187 1188 1189 1190 1191 1192 1193 1194 1195 1196 1197 1198 1199 1200 1201 1202 1203 1204 1205 1206 1207 1208 1209 1210 1211 1212 1213 1214 1215 1216 1217 1218 1219 1220 1221 1222 1223 1224 1225 1226 1227 1228 1229 1230 1231 1232 1233 1234 1235 1236 1237 1238 1239 1240 1241 1242 1243 1244 1245 1246 1247 1248 1249 1250 1251 1252 1253 1254 1255 1256 1257 1258 1259 1260 1261 1262 1263 1264 1265 1266 1267 1268 1269 1270 1271 1272 1273 1274 1275 1276 1277 1278 1279 1280 1281 1282 1283 1284 1285 1286 1287 1288 1289 1290 1291 1292 1293 1294 1295 1296 1297 1298 1299 1300 1301 1302 1303 1304 1305 1306 1307 1308 1309 1310 1311 1312 1313 1314 1315 1316 1317 1318 1319 1320 1321 1322 1323 1324 1325 1326 1327 1328 1329 1330 1331 1332 1333 1334 1335 1336 1337 1338 1339 1340 1341 1342 1343 1344 1345 1346 1347 1348 1349 1350 1351 1352 1353 1354 1355 1356 1357 1358 1359 1360 1361 1362 1363 1364 1365 1366 1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403 1404 1405 1406 1407 1408 1409 1410 1411 1412 1413 1414 1415 1416 1417 1418 1419 1420 1421 1422 1423 1424 1425 1426 1427 1428 1429 1430 1431 1432 1433 1434 1435 1436 1437 1438 1439 1440 1441 1442 1443 1444 1445 1446 1447 1448 1449 1450 1451 1452 1453 1454 1455 1456 1457 1458 1459 1460 1461 1462 1463 1464 1465 1466 1467 1468 1469 1470 1471 1472 1473 1474 1475 1476 1477 1478 1479 1480 1481 1482 1483 1484 1485 1486 1487 1488 1489 1490 1491 1492 1493 1494 1495 1496 1497 1498 1499 1500 1501 1502 1503 1504 1505 1506 1507 1508 1509 1510 1511 1512 1513 1514 1515 1516 1517 1518 1519 1520 1521 1522 1523 1524 1525 1526 1527 1528 1529 1530 1531 1532 1533 1534 1535 1536 1537 1538 1539 1540 1541 1542 1543 1544 1545 1546 1547 1548 1549 1550 1551 1552 1553 1554 1555 1556 1557 1558 1559 1560 1561 1562 1563 1564 1565 1566 1567 1568 1569 1570 1571 1572 1573 1574 1575 1576 1577 1578 1579 1580 1581 1582 1583 1584 1585 1586 1587 1588 1589 1590 1591 1592 1593 1594 1595 1596 1597 1598 1599 1600 1601 1602 1603 1604 1605 1606 1607 1608 1609 1610 1611 1612 1613 1614 1615 1616 1617 1618 1619 1620 1621 1622 1623 1624 1625 1626 1627 1628 1629 1630 1631 1632 1633 1634 1635 1636 1637 1638 1639 1640 1641 1642 1643 1644 1645 1646 1647 1648 1649 1650 1651 1652 1653 1654 1655 1656 1657 1658 1659 1660 1661 1662 1663 1664 1665 1666 1667 1668 1669 1670 1671 1672 1673 1674 1675 1676 1677 1678 1679 1680 1681 1682 1683 1684 1685 1686 1687 1688 1689 1690 1691 1692 1693 1694 1695 1696 1697 1698 1699 1700 1701 1702 1703 1704 1705 1706 1707 1708 1709 1710 1711 1712 1713 1714 1715 1716 1717 1718 1719 1720 1721 1722 1723 1724 1725 1726 1727 1728 1729 1730 1731 1732 1733 1734 1735 1736 1737 1738 1739 1740 1741 1742 1743 1744 1745 1746 1747 1748 1749 1750 1751 1752 1753 1754 1755 1756 1757 1758 1759 1760 1761 1762 1763 1764 1765 1766 1767 1768 1769 1770 1771 1772 1773 1774 1775 1776 1777 1778 1779 1780 1781 1782 1783 1784 1785 1786 1787 1788 1789 1790 1791 1792 1793 1794 1795 1796 1797 1798 1799 1800 1801 1802 1803 1804 1805 1806 1807 1808 1809 1810 1811 1812 1813 1814 1815 1816 1817 1818 1819 1820 1821 1822 1823 1824 1825 1826 1827 1828 1829 1830 1831 1832 1833 1834 1835 1836 1837 1838 1839 1840 1841 1842 1843 1844 1845 1846 1847 1848 1849 1850 1851 1852 1853 1854 1855 1856 1857 1858 1859 1860 1861 1862 1863 1864 1865 1866 1867 1868 1869 1870 1871 1872 1873 1874 1875 1876 1877 1878 1879 1880 1881 1882 1883 1884 1885 1886 1887 1888 1889 1890 1891 1892 1893 1894 1895 1896 1897 1898 1899 1900 1901 1902 1903 1904 1905 1906 1907 1908 1909 1910 1911 1912 1913 1914 1915 1916 1917 1918 1919 1920 1921 1922 1923 1924 1925 1926 1927 1928 1929 1930 1931 1932 1933 1934 1935 1936 1937 1938 1939 1940 1941 1942 1943 1944 1945 1946 1947 1948 1949 1950 1951 1952 1953 1954 1955 1956 1957 1958 1959 1960 1961 1962 1963 1964 1965 1966 1967 1968 1969 1970 1971 1972 1973 1974 1975 1976 1977 1978 1979 1980 1981 1982 1983 1984 1985 1986 1987 1988 1989 1990 1991 1992 1993 1994 1995 1996 1997 1998 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012 2013 2014 2015 2016 2017 2018 2019 2020 2021 2022 2023 2024 2025 2026 2027 2028 2029 2030 2031 2032 2033 2034 2035 2036 2037 2038 2039 2040 2041 2042 2043 2044 2045 2046 2047 2048 2049 2050 2051 2052 2053 2054 2055 2056 2057 2058 2059 2060 2061 2062 2063 2064 2065 2066 2067 2068 2069 2070 2071 2072 2073 2074 2075 2076 2077 2078 2079 2080 2081 2082 2083 2084 2085 2086 2087 2088 2089 2090 2091 2092 2093 2094 2095 2096 2097 2098 2099 2100 2101 2102 2103 2104 2105 2106 2107 2108 2109 2110 2111 2112 2113 2114 2115 2116 2117 2118 2119 2120 2121 2122 2123 2124 2125 2126 2127 2128 2129 2130 2131 2132 2133 2134 2135 2136 2137 2138 2139 2140 2141 2142 2143 2144 2145 2146 2147 2148 2149 2150 2151 2152 2153 2154 2155 2156 2157 2158 2159 2160 2161 2162 2163 2164 2165 2166 2167 2168 2169 2170 2171 2172 2173 2174 2175 2176 2177 2178 2179 2180 2181 2182 2183 2184 2185 2186 2187 2188 2189 2190 2191 2192 2193 2194 2195 2196 2197 2198 2199 2200 2201 2202 2203 2204 2205 2206 2207 2208 2209 2210 2211 2212 2213 2214 2215 2216 2217 2218 2219 2220 2221 2222 2223 2224 2225 2226 2227 2228 2229 2230 2231 2232 2233 2234 2235 2236 2237 2238 2239 2240 2241 2242 2243 2244 2245 2246 2247 2248 2249 2250 2251 2252 2253 2254 2255 2256 2257 2258 2259 2260 2261 2262 2263 2264 2265 2266 2267 2268 2269 2270 2271 2272 2273 2274 2275 2276 2277 2278 2279 2280 2281 2282 2283 2284 2285 2286 2287 2288 2289 2290 2291 2292 2293 2294 2295 2296 2297 2298 2299 2300 2301 2302 2303 2304 2305 2306 2307 2308 2309 2310 2311 2312 2313 2314 2315 2316 2317 2318 2319 2320 2321 2322 2323 2324 2325 2326 2327 2328 2329 2330 2331 2332 2333 2334 2335 2336 2337 2338 2339 2340 2341 2342 2343 2344 2345 2346 2347 2348 2349 2350 2351 2352 2353 2354 2355 2356 2357 2358 2359 2360 2361 2362 2363 2364 2365 2366 2367 2368 2369 2370 2371 2372 2373 2374 2375 2376 2377 2378 2379 2380 2381 2382 2383 2384 2385 2386 2387 2388 2389 2390 2391 2392 2393 2394 2395 2396 2397 2398 2399 2400 2401 2402 2403 2404 2405 2406 2407 2408 2409 2410 2411 2412 2413 2414 2415 2416 2417 2418 2419 2420 2421 2422 2423 2424 2425 2426 2427 2428 2429 2430 2431 2432 2433 2434 2435 2436 2437 2438 2439 2440 2441 2442 2443 2444 2445 2446 2447 2448 2449 2450 2451 2452 2453 2454 2455 2456 2457 2458 2459 2460 2461 2462 2463 2464 2465 2466 2467 2468 2469 2470 2471 2472 2473 2474 2475 2476 2477 2478 2479 2480 2481 2482 2483 2484 2485 2486 2487 2488 2489 2490 2491 2492 2493 2494 2495 2496 2497 2498 2499 2500 2501 2502 2503 2504 2505 2506 2507 2508 2509 2510 2511 2512 2513 2514 2515 2516 2517 2518 2519 2520 2521 2522 2523 2524 2525 2526 2527 2528 2529 2530 2531 2532 2533 2534 2535 2536 2537 2538 2539 2540 2541 2542 2543 2544 2545 2546 2547 2548 2549 2550 2551 2552 2553 2554 2555 2556 2557 2558 2559 2560 2561 2562 2563 2564 2565 2566 2567 2568 2569 2570 2571 2572 2573 2574 2575 2576 2577 2578 2579 2580 2581 2582 2583 2584 2585 2586 2587 2588 2589 2590 2591 2592 2593 2594 2595 2596 2597 2598 2599 2600 2601 2602 2603 2604 2605 2606 2607 2608 2609 2610 2611 2612 2613 2614 2615 2616 2617 2618 2619 2620 2621 2622 2623 2624 2625 2626 2627 2628 2629 2630 2631 2632 2633 2634 2635 2636 2637 2638 2639 2640 2641 2642 2643 2644 2645 2646 2647 2648 2649 2650 2651 2652 2653 2654 2655 2656 2657 2658 2659 2660 2661

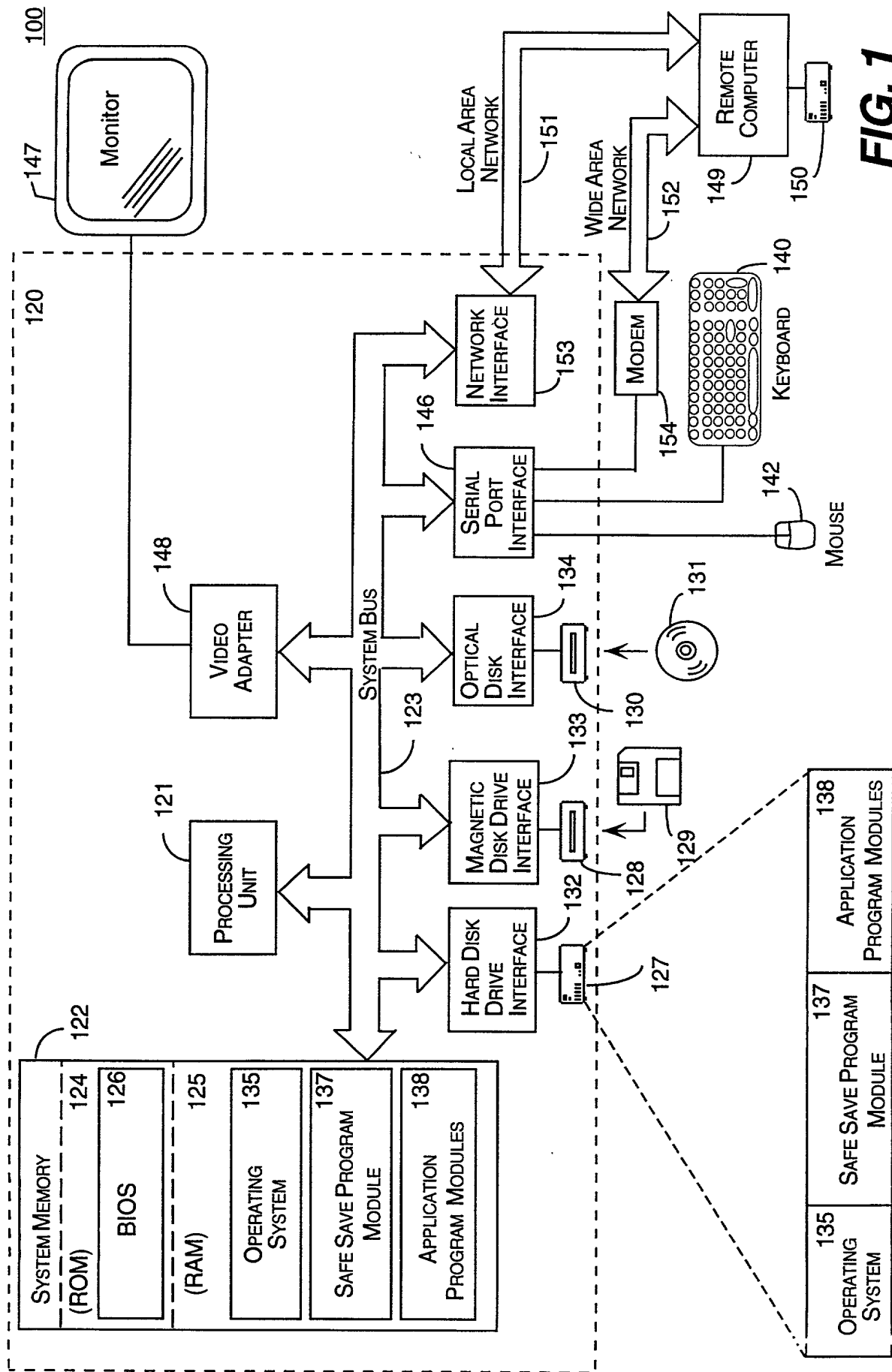
## ABSTRACT

### SAFE SAVE METHOD FOR HTML DOCUMENTS

5 A computer-implemented process for replacing old files having old  
names on a permanent storage medium with new files in volatile memory.  
Some of these files may be opaque files, which are old files that have become  
new files without undergoing any changes, other than possibly name changes.  
The primary steps of this replacement can comprise first saving the non-  
10 opaque new files onto the permanent storage medium and assigning them  
temporary filenames. An opaque recovery file listing the old name and a  
temporary name for each opaque file is created. A primary recovery file  
listing the temporary name and a final name for each new file is created.  
After the names of the opaque files are changed to their temporary names,  
15 the names of the non-opaque old files are changed from their old names to  
backup names. Then, the names of the new files are changed from their  
temporary names to their final names. The recovery files and the old files  
can be deleted. If an event leaves the program performing the primary steps  
in control but unable to complete the replacement, then the old files can be  
20 restored to the permanent storage medium. If an event prevents completion  
of either a replacement attempt or a restoration of the old files, completion  
of the replacement can again be attempted at a later time using the recovery  
files. The described processes can be applied to saving HTML documents,  
each of which typically comprises multiple files.

25 J&A Docket No. 13237-2390

MS Docket No. 116280.1



**FIG. 1**

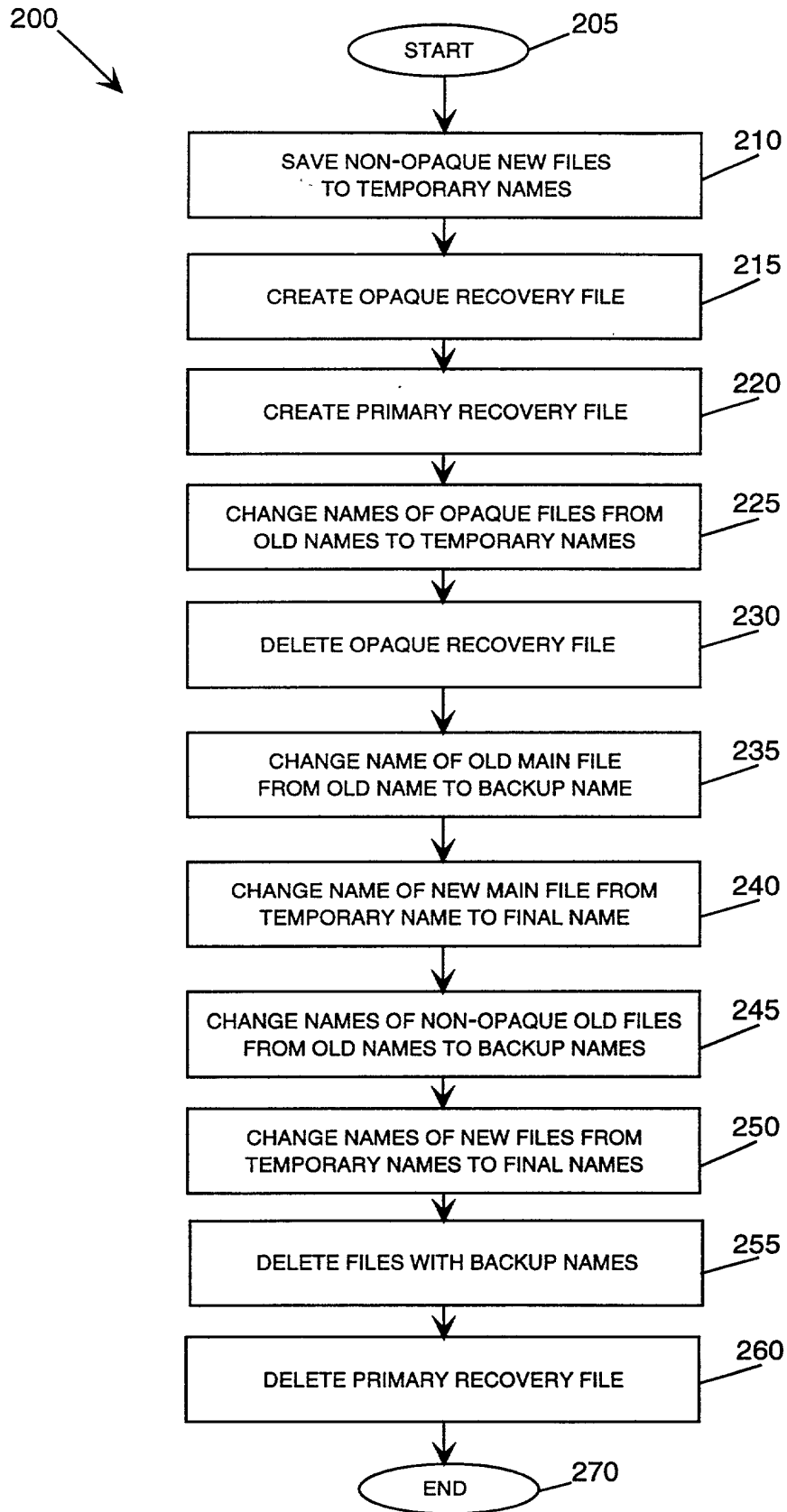


FIG. 2



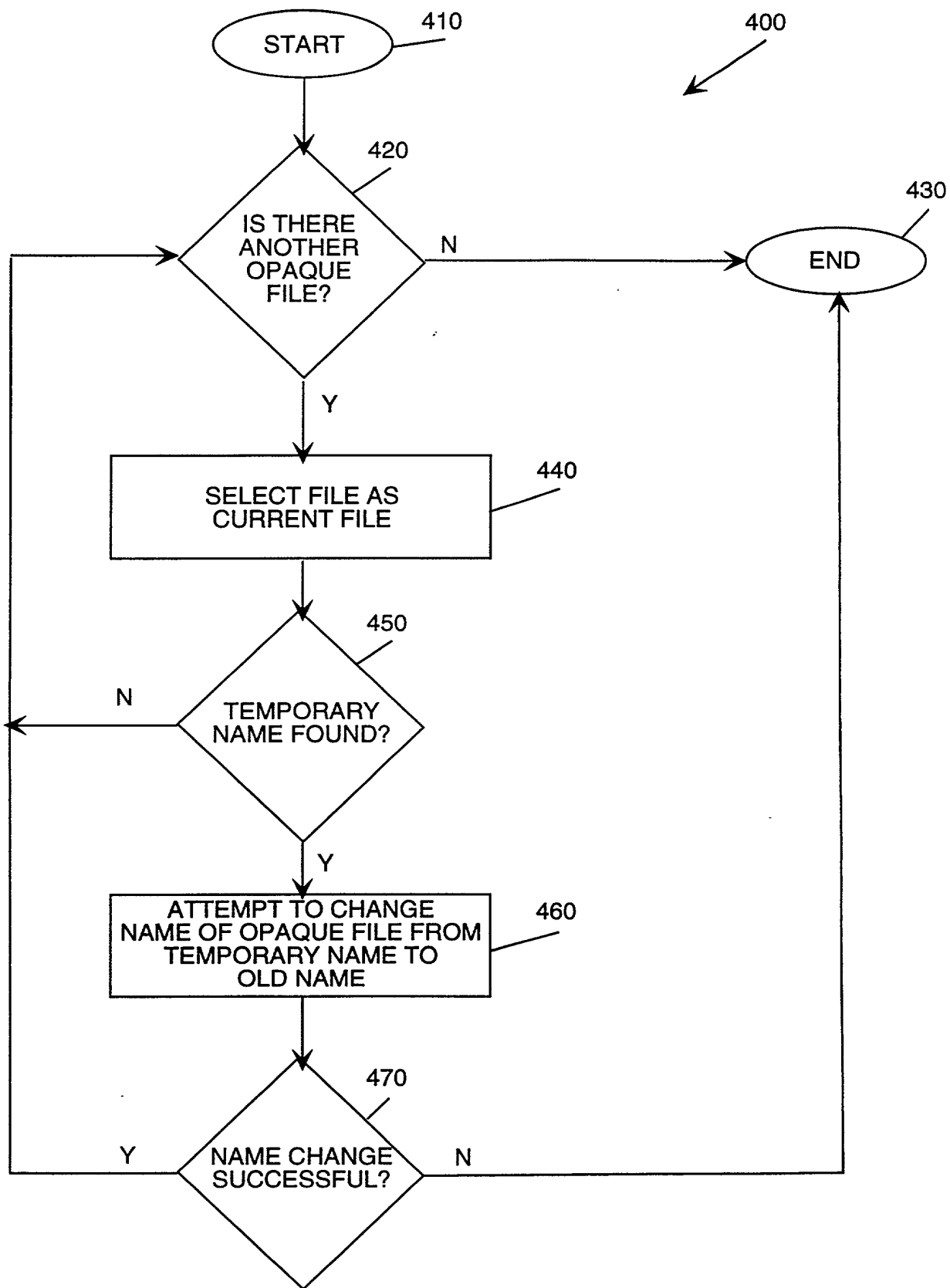


FIG. 4

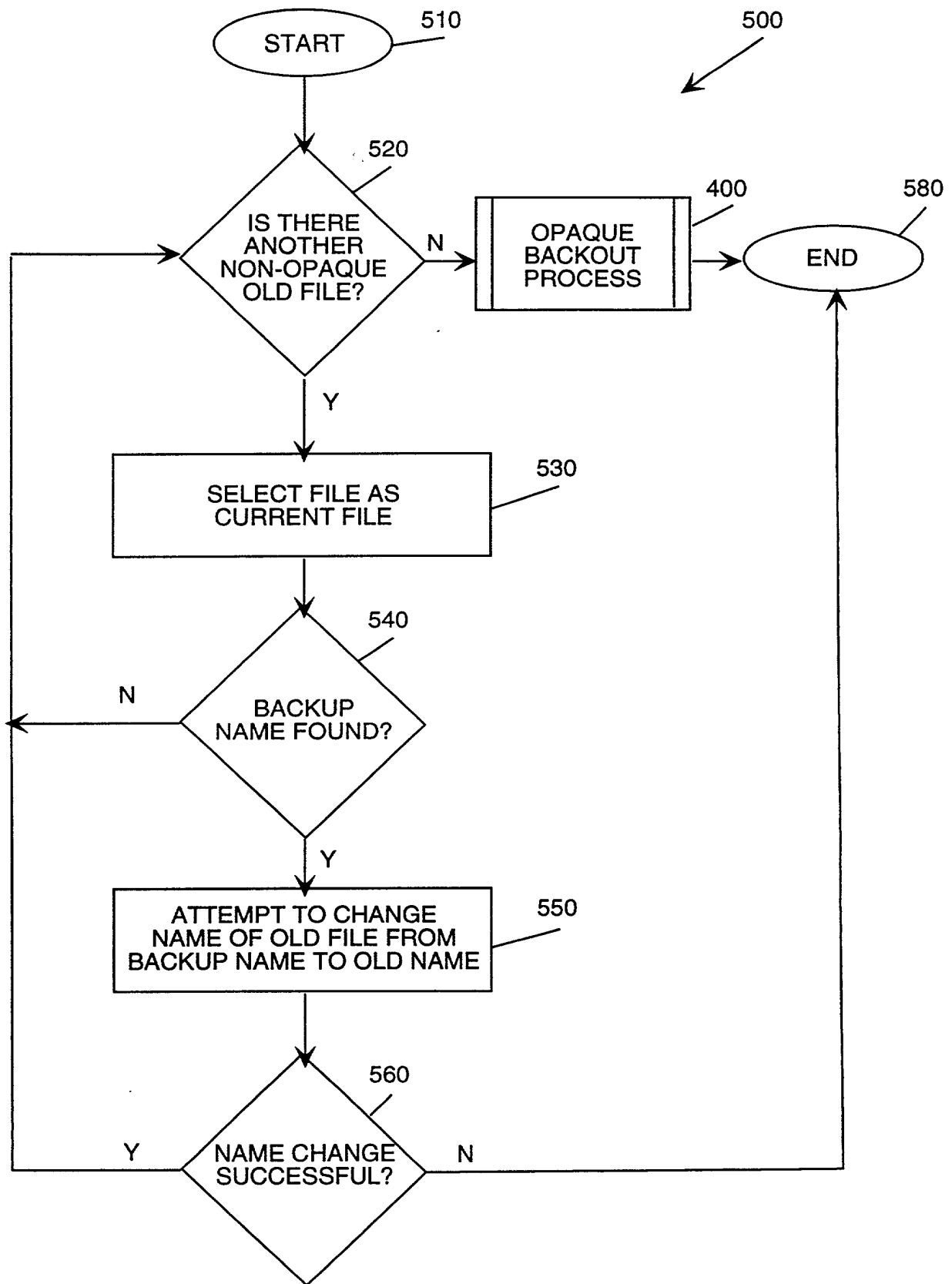
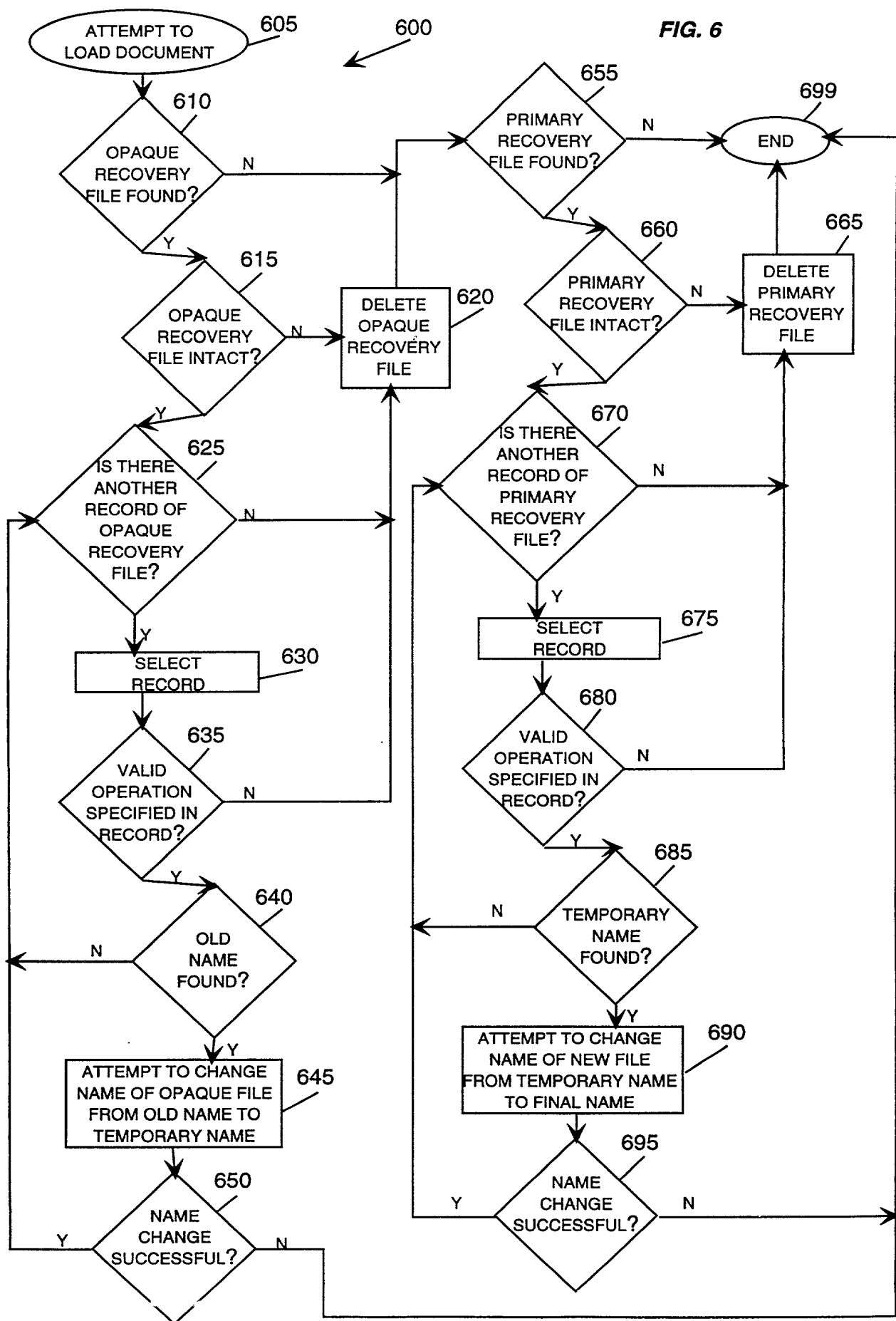


FIG. 5





**DECLARATION AND POWER OF ATTORNEY**

Attorney's Docket No. 13237-2390/MS 116280.1

In re application of Alan A. Ramaley, Thomas W. Olsen, Darrell L. Aldrich, and David M. Buchthal, as a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name. I believe I am an original, first and joint inventor of the subject matter which is claimed and for which a patent is sought on the invention entitled: "**SAFE SAVE METHOD FOR HTML DOCUMENTS**", the specification of which is filed herewith.

I hereby state that I have reviewed and understand the contents of the above-identified specification, including the claims, as amended by any amendment referred to above. I do not know and do not believe that the same was ever known or used by others in the United States of America before my or our invention thereof, or patented or described in any printed publication in any country before my or our invention thereof or more than one year prior to the date of this application. I further state that the invention was not in public use or on sale in the United States of America more than one year prior to the date of this application. *I understand that I have a duty of candor and good faith toward the Patent and Trademark Office*, and I acknowledge the duty to disclose information which is material to the examination of this application in accordance with Title 37, Code of Federal Regulations, §1.56.

I hereby claim foreign priority benefits under Title 35, United States Code, §119 (a)-(d) of the foreign application(s) for patent or inventor's certificate listed below, and have also identified below any foreign application for patent or inventor's certificate disclosing subject matter in common with the above-identified specification and having a filing date before that of the application on which priority is claimed:

<u>Application No.</u>	<u>Country</u>	<u>Filing Date</u>	<u>Priority Claimed Under 35 USC §119</u>
none			Yes _____ No <u>X</u>

I hereby claim the benefit under Title 35, United States Code, § 119(e) of any United States provisional application(s) listed below:

<u>none</u>			
(Application No.)	(Filing Date)	(Application No.)	(Filing Date)

I hereby claim the benefit under Title 35, United States Code, §120 of any United States application(s) listed below and, insofar as the subject matter disclosed and claimed in the present application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code §112, I acknowledge the duty to disclose material information as defined in Title 37, Code of Federal Regulations, §1.56 which became available between the filing date of the prior application and the national or PCT international filing date of this application:

<u>Application Serial No.</u>	<u>Filing Date</u>	<u>Status: patented, pending, abandoned</u>
none		

I further declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statement were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code, and that such willful false statements may jeopardize the validity of the application or any patents issuing thereon.

**POWER OF ATTORNEY:** The following attorneys are hereby appointed to prosecute this application and transact all business in the Patent and Trademark Office connected therewith: Anthony B. Askew - 24,154; Roger T. Frost - 22,176; Jeffrey E. Young - 28,490; Robert E. Richards - 29,105; John R. Harris - 30,388; Stephen M. Schaetzel -31,418; Larry A. Roberts - 31,871; ~~Thomas A. Hodge - 22,602~~; Gregory T. Gronholm - 32,415; Dale Lischer - 28,438; Peter G. Pappas - 33,205; James Dean Johnson - 31,771; W. Scott Petty - 35,645; Daniel J. Warren - 34,272; Leona G. Young - 37,266; Jamie L. Greene-32,467; Holmes J. Hawkins III - 38,913; Mary Anthony Merchant - 39,771; Michael J. Mehrman - 40,086; William L. Warren - 36,714; Brenda Ozaki Holmes - 40,339; James D. Withers - 40,376; ~~M. Todd Mitchem - 40,731~~; Kimberly J. Prior - 41,483; ~~William B. Lafferty - 39,259~~; Theodore M. Green - 41,801; Christopher J. Leonard - 41,940; Christos S. Kyriakou - 42,776; John K. McDonald - 42,860; Michael S. Pavento - 42,985; Suzanne Seavello Shope - 37,933; M. Scott Boone - 42,341; Sima Singadia Kulkarni - P43,732; A. Shane Nichols - P43,836; ~~Robert J. O'Connell - P44,265~~; Christopher J. Chan - P44,070; Katie E. Sako - 32,628; Daniel D. Crouse - 32,022.

Send correspondence to: **JONES & ASKEW, LLP**  
2400 Monarch Tower, 3424 Peachtree Road, N.E.  
Atlanta, GA 30326

Direct telephone calls at (404) 949-2400

**W. Scott Petty**

Full name of first joint inventor: Alan A. Ramaley	Citizenship: U.S.
Inventor's signature	Date:
Residence and Post Office Address: 704 E. Thomas Street, Apt. 201, Seattle, Washington 98102	

**DECLARATION AND POWER OF ATTORNEY**

Attorney's Docket No. 13237-2390/MS 116280.1

In re application of Alan A. Ramaley, Thomas W. Olsen, Darrell L. Aldrich, and David M. Buchthal, as a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name. I believe I am an original, second and joint inventor of the subject matter which is claimed and for which a patent is sought on the invention entitled: "**SAFE SAVE METHOD FOR HTML DOCUMENTS**", the specification of which is filed herewith.

I hereby state that I have reviewed and understand the contents of the above-identified specification, including the claims, as amended by any amendment referred to above. I do not know and do not believe that the same was ever known or used by others in the United States of America before my or our invention thereof, or patented or described in any printed publication in any country before my or our invention thereof or more than one year prior to the date of this application. I further state that the invention was not in public use or on sale in the United States of America more than one year prior to the date of this application. *I understand that I have a duty of candor and good faith toward the Patent and Trademark Office*, and I acknowledge the duty to disclose information which is material to the examination of this application in accordance with Title 37, Code of Federal Regulations, §1.56.

I hereby claim foreign priority benefits under Title 35, United States Code, §119 (a)-(d) of the foreign application(s) for patent or inventor's certificate listed below, and have also identified below any foreign application for patent or inventor's certificate disclosing subject matter in common with the above-identified specification and having a filing date before that of the application on which priority is claimed:

<u>Application No.</u>	<u>Country</u>	<u>Filing Date</u>	<u>Priority Claimed Under 35 USC §119</u>
<u>none</u>			Yes _____ No <u>X</u>

I hereby claim the benefit under Title 35, United States Code, § 119(e) of any United States provisional application(s) listed below:

<u>Application No.</u>	<u>Filing Date</u>	<u>Application No.</u>	<u>Filing Date</u>
<u>none</u>			

I hereby claim the benefit under Title 35, United States Code, §120 of any United States application(s) listed below and, insofar as the subject matter disclosed and claimed in the present application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code §112, I acknowledge the duty to disclose material information as defined in Title 37, Code of Federal Regulations, §1.56 which became available between the filing date of the prior application and the national or PCT international filing date of this application:

<u>Application Serial No.</u>	<u>Filing Date</u>	<u>Status: patented, pending, abandoned</u>
<u>none</u>		

I further declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code, and that such willful false statements may jeopardize the validity of the application or any patents issuing thereon.

**POWER OF ATTORNEY:** The following attorneys are hereby appointed to prosecute this application and transact all business in the Patent and Trademark Office connected therewith: Anthony B. Askew - 24,154; Roger T. Frost - 22,176; Jeffrey E. Young - 28,490; Robert E. Richards - 29,105; John R. Harris - 30,388; Stephen M. Schaetzel - 31,418; Larry A. Roberts - 31,871; ~~Thomas A. Hodge, 22,602~~; Gregory T. Gronholm - 32,415; Dale Lischer - 28,438; Peter G. Pappas - 33,205; James Dean Johnson - 31,771; W. Scott Petty - 35,645; Daniel J. Warren - 34,272; Leona G. Young - 37,266; Jamie L. Greene - 32,467; Holmes J. Hawkins III - 38,913; Mary Anthony Merchant - 39,771; Michael J. Mehrman - 40,086; William L. Warren - 36,714; Brenda Ozaki Holmes - 40,339; James D. Withers - 40,376; ~~M. Todd Mitchem - 40,731~~; Kimberly J. Prior - 41,483; ~~William D. Lafferty - 39,259~~; Theodore M. Green - 41,801; Christopher J. Leonard - 41,940; Christos S. Kyriakou - 42,776; John K. McDonald - 42,860; Michael S. Pavento - 42,985; Suzanne Seavello Shope - 37,933; M. Scott Boone - 42,341; Sima Singadia Kulkarni - P43,732; A. Shane Nichols - P43,836; ~~Robert J. O'Connell - P44,265~~; Christopher J. Chan - P44,070; Katie E. Sako - 32,628; Daniel D. Crouse - 32,022.

Send correspondence to: **JONES & ASKEW, LLP**  
2400 Monarch Tower, 3424 Peachtree Road, N.E.  
Atlanta, GA 30326

Direct telephone calls at **(404) 949-2400**  
**W. Scott Petty**

Full name of first joint inventor: Thomas W. Olsen	Citizenship: U.S.
Inventor's signature	Date:
Residence and Post Office Address: 4031 251 <sup>st</sup> Place, S.E., Issaquah, Washington 98029	

**DECLARATION AND POWER OF ATTORNEY**Attorney's Docket No. **13237-2390/MS 116280.1**

In re application of Alan A. Ramaley, Thomas W. Olsen, Darrell L. Aldrich, and David M. Buchthal, as a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name. I believe I am an original, third and joint inventor of the subject matter which is claimed and for which a patent is sought on the invention entitled: "**SAFE SAVE METHOD FOR HTML DOCUMENTS**", the specification of which is filed herewith.

I hereby state that I have reviewed and understand the contents of the above-identified specification, including the claims, as amended by any amendment referred to above. I do not know and do not believe that the same was ever known or used by others in the United States of America before my or our invention thereof, or patented or described in any printed publication in any country before my or our invention thereof or more than one year prior to the date of this application. I further state that the invention was not in public use or on sale in the United States of America more than one year prior to the date of this application. *I understand that I have a duty of candor and good faith toward the Patent and Trademark Office*, and I acknowledge the duty to disclose information which is material to the examination of this application in accordance with Title 37, Code of Federal Regulations, §1.56.

I hereby claim foreign priority benefits under Title 35, United States Code, §119 (a)-(d) of the foreign application(s) for patent or inventor's certificate listed below, and have also identified below any foreign application for patent or inventor's certificate disclosing subject matter in common with the above-identified specification and having a filing date before that of the application on which priority is claimed:

<u>Application No.</u>	<u>Country</u>	<u>Filing Date</u>	<u>Priority Claimed Under 35 USC §119</u>
none			Yes _____ No <u>X</u>

I hereby claim the benefit under Title 35, United States Code, § 119(e) of any United States provisional application(s) listed below:

<u>none</u>			
(Application No.)	(Filing Date)	(Application No.)	(Filing Date)

I hereby claim the benefit under Title 35, United States Code, §120 of any United States application(s) listed below and, insofar as the subject matter disclosed and claimed in the present application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code §112, I acknowledge the duty to disclose material information as defined in Title 37, Code of Federal Regulations, §1.56 which became available between the filing date of the prior application and the national or PCT international filing date of this application:

<u>Application Serial No.</u>	<u>Filing Date</u>	<u>Status: patented, pending, abandoned</u>
none		

I further declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statement were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code, and that such willful false statements may jeopardize the validity of the application or any patents issuing thereon.

**POWER OF ATTORNEY:** The following attorneys are hereby appointed to prosecute this application and transact all business in the Patent and Trademark Office connected therewith: Anthony B. Askew - 24,154; Roger T. Frost - 22,176; Jeffrey E. Young - 28,490; Robert E. Richards - 29,105; John R. Harris - 30,388; Stephen M. Schaetzel - 31,418; Larry A. Roberts - 31,871; ~~Thomas A. Hodge - 22,602~~; Gregory T. Gronholm - 32,415; Dale Lischer - 28,438; Peter G. Pappas - 33,205; James Dean Johnson - 31,771; W. Scott Petty - 35,645; Daniel J. Warren - 34,272; Leona G. Young - 37,266; Jamie L. Greene - 32,467; Holmes J. Hawkins III - 38,913; Mary Anthony Merchant - 39,771; Michael J. Mehrman - 40,086; William L. Warren - 36,714; Brenda Ozaki Holmes - 40,339; James D. Withers - 40,376; ~~M. Todd Mitchem - 40,731~~; Kimberly J. Prior - 41,483; ~~William B. Lafferty - 39,259~~; Theodore M. Green - 41,801; Christopher J. Leonard - 41,940; Christos S. Kyriakou - 42,776; John K. McDonald - 42,860; Michael S. Pavento - 42,985; Suzanne Seavello Shope - 37,933; M. Scott Boone - 42,341; Sima Singadia Kulkarni - P43,732; A. Shane Nichols - P43,836; ~~Robert J. O'Connell - P44,265~~; Christopher J. Chan - P44,070; Katie E. Sako - 32,628; Daniel D. Crouse - 32,022.

Send correspondence to: **JONES & ASKEW, LLP**  
2400 Monarch Tower, 3424 Peachtree Road, N.E.  
Atlanta, GA 30326

Direct telephone calls at **(404) 949-2400**

**W. Scott Petty**

Full name of first joint inventor: Darrell L. Aldrich	Citizenship: U.S.
Inventor's signature	Date:
Residence and Post Office Address: 20708 N.E. 90 <sup>th</sup> Street, Redmond, Washington 98052	

**DECLARATION AND POWER OF ATTORNEY**Attorney's Docket No. **13237-2390/MS 116280.1**

In re application of Alan A. Ramaley, Thomas W. Olsen, Darrell L. Aldrich, and David M. Buchthal, as a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name. I believe I am an original, fourth and joint inventor of the subject matter which is claimed and for which a patent is sought on the invention entitled: "**SAFE SAVE METHOD FOR HTML DOCUMENTS**", the specification of which is filed herewith.

I hereby state that I have reviewed and understand the contents of the above-identified specification, including the claims, as amended by any amendment referred to above. I do not know and do not believe that the same was ever known or used by others in the United States of America before my or our invention thereof, or patented or described in any printed publication in any country before my or our invention thereof or more than one year prior to the date of this application. I further state that the invention was not in public use or on sale in the United States of America more than one year prior to the date of this application. *I understand that I have a duty of candor and good faith toward the Patent and Trademark Office*, and I acknowledge the duty to disclose information which is material to the examination of this application in accordance with Title 37, Code of Federal Regulations, §1.56.

I hereby claim foreign priority benefits under Title 35, United States Code, §119 (a)-(d) of the foreign application(s) for patent or inventor's certificate listed below, and have also identified below any foreign application for patent or inventor's certificate disclosing subject matter in common with the above-identified specification and having a filing date before that of the application on which priority is claimed:

<u>Application No.</u>	<u>Country</u>	<u>Filing Date</u>	<u>Priority Claimed Under 35 USC §119</u>
<u>none</u>			Yes _____ No <u>X</u>

I hereby claim the benefit under Title 35, United States Code, § 119(e) of any United States provisional application(s) listed below:

<u>none</u>			
<u>(Application No.)</u>	<u>(Filing Date)</u>	<u>(Application No.)</u>	<u>(Filing Date)</u>

I hereby claim the benefit under Title 35, United States Code, §120 of any United States application(s) listed below and, insofar as the subject matter disclosed and claimed in the present application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code §112, I acknowledge the duty to disclose material information as defined in Title 37, Code of Federal Regulations, §1.56 which became available between the filing date of the prior application and the national or PCT international filing date of this application:

<u>Application Serial No.</u>	<u>Filing Date</u>	<u>Status: patented, pending, abandoned</u>
<u>none</u>		

I further declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code, and that such willful false statements may jeopardize the validity of the application or any patents issuing thereon.

**POWER OF ATTORNEY:** The following attorneys are hereby appointed to prosecute this application and transact all business in the Patent and Trademark Office connected therewith: Anthony B. Askew - 24,154; Roger T. Frost - 22,176; Jeffrey E. Young - 28,490; Robert E. Richards - 29,105; John R. Harris - 30,388; Stephen M. Schaetzel - 31,418; Larry A. Roberts - 31,871; ~~Thomas A. Hodge - 22,602~~; Gregory T. Gronholm - 32,415; Dale Lischer - 28,438; Peter G. Pappas - 33,205; James Dean Johnson - 31,771; W. Scott Petty - 35,645; Daniel J. Warren - 34,272; Leona G. Young - 37,266; Jamie L. Greene - 32,467; Holmes J. Hawkins III - 38,913; Mary Anthony Merchant - 39,771; Michael J. Mehrman - 40,086; William L. Warren - 36,714; Brenda Ozaki Holmes - 40,339; James D. Withers - 40,376; ~~M. Todd Mitchem - 40,734~~; Kimberly J. Prior - 41,483; ~~William B. Lafferty - 39,259~~; Theodore M. Green - 41,801; Christopher J. Leonard - 41,940; Christos S. Kyriakou - 42,776; John K. McDonald - 42,860; Michael S. Pavento - 42,985; Suzanne Seavello Shope - 37,933; M. Scott Boone - 42,341; Sima Singadia Kulkarni - P43,732; A. Shane Nichols - P43,836; ~~Robert J. O'Connell - P44,265~~; Christopher J. Chan - P44,070; Katie E. Sako - 32,628; Daniel D. Crouse - 32,022.

Send correspondence to: **JONES & ASKEW, LLP**  
2400 Monarch Tower, 3424 Peachtree Road, N.E.  
Atlanta, GA 30326

Direct telephone calls at **(404) 949-2400**  
**W. Scott Petty**

Full name of first joint inventor: David M. Buchthal	Citizenship: U.S.
Inventor's signature	Date:
Residence and Post Office Address: 15408 278 <sup>th</sup> Avenue, N.E., Duvall, Washington 98019	